**房屋租赁系统的设计与实现**

**摘 要**：随着近几年经济的快速增长，房价大幅上涨，对于刚刚毕业的大学生而言，一套体面的房子是支付不起的，所以大多数年轻人选择租房，基于当前的环境现状，这次的毕业论文打算设计一套线上的房屋租赁系统。本系统基于以上需求,设计一个在线房屋租赁系统,租房用户可以在系统中搜索符合自己需求的房源，同时帮助出租户发布房屋出租信息。系统管理员在后台批量管理各种信息,可大量的节省工作时间，提高工作的效率。

本系统采用Javaweb，Mysql，Html，Css，Jquery,Servlet等知识来实现该系统，其中前端以Html5，Css来设计网页，后端通过Servlet控制，数据库用Mysql来实现，用IDEA来作为开发工具。

**关键词**：房屋租赁系统、Servlet、数据库开发；

**Design and Implementation of House Rental System**

**Abstract**：With the rapid economic growth in recent years, the house price has risen sharply. For the college students who have just graduated, a decent house is not affordable, so most young people choose to rent a house. Based on the current environmental status, this graduation thesis plans to design an online house rental system. This system is based on the above needs, design an online housing rental system, rental users can search in the system to meet their own needs of the housing source, at the same time to help tenants release housing rental information. System administrators in the background batch management of various information, can save a lot of work time, improve the efficiency of work.

This system uses JavaWeb, Mysql, HTML, CSS, jQuery, Servlet and other knowledge to achieve the system,The front end is designed by HTML5 and CSS, the back end is controlled by Servlet, and the database is realized by MYSQL IDEA as a development tool.

**Keywords:** Housing Rental system, Servlet, Database Development;

# 第1章 课题的研究背景

## 1.1 课题的研究背景

近年来,随着国科学技术研究领域不断深入，国社会科学技术水平不断突破，经济快速增长,现代人对于房屋的质量需求也在不断增多，而从中占有比较大的一部分的是房屋出租这一方面。在房屋租赁企业中面对繁琐而又庞大的信息量,为了企业和人们可以更好的了解和最快知道最新的房源信息。这次的系统设计为了解决相关的问题,打算设计一个关于房屋出租的系统，现代的租房用户可以按照自己的需求来寻找适合自己的房源，同样的如果房屋供应商有剩余的房源，也可以在这个网站上发布。而对于系统管理员来说，只需在后台将供应商提供的房源发布出去,以及应对租客的相关信息，管理租房看房的相关日常即可，这样的话就能大大的节省时间，提高房屋管理员的工作效率。主要来说，房屋租赁管理非常重要，是确保线上房屋租赁正常运行的关键和基础，在互联网环境下，有必要在过程中引入互联网技术,实现信息化科学化智能化，提高房屋租赁管理效率与质量，并且对现有房屋的质量进行严格管控，提升服务质量，提高消费者对房屋质量的满意度。将房屋管理系统运用到现有的租赁市场中去，可以为传统行业带来新的发展前景，扩宽现有的销售模式，扩大房屋销售的空间，并通过信息化的手段为相关的房东出租租客租房重要业务提供很好的技术支持。一个简介，美观功能全面的房屋租赁系统对于以房屋出租买卖的市场人员来说无疑是一个非常巨大的帮助。本文设计的房屋租赁系统实现了房屋出租屋信息的发布，租客租房的信息，以及一些常见的管理等功能。

为了不被社会经济迅猛发展的浪潮淹没，同时提高房屋租赁的效率，改善当前的消费模式，相对于房屋出租的房东而言，一些复杂的租金合同，以及房产信息在现代很难通过传统的文字合同进行管理;对于需要房屋的现代年轻人而言，各种各样的房屋信息对他们来说是一个不小的头疼问题。此时一个功能强大的房屋租赁管理系统就变得尤为重要,此次系统是在经过大量的市场调研以及研究现有的市场模式诞生的,从当代人们的现有需求出发,采用以Servlet技术和idea作为开发工具，在配置好相关环境后，最终设计了一个功能强大，界面简介的房屋租赁管理网站。对于房屋管理员来说，房屋租赁意味着对租客以及房东的相关合同信息进行管理不仅如此，还要对客户的资料，合同租金，房屋出租信息进行严谨的管理，这是一项重大且复杂的工作。如果采用人工管理的话会产生一些不可避免的错误，比如数据丢失，而且还会占据大量的时间和精力，对于时间就是金钱的21世纪来说，这是得不偿失的。如果让电脑来代替人工进行房屋租赁的管理，一方面可以节省掉人工大量的时间，另一方面还具有能

保证数据精确无误，保存时间长的优点。采用计算机来代替人工进行管理操作还具有很多的优点.例如:使用时间长，查找速度快，防止数据泄露，工作量大且质量好等。这些计算机的优点正是科学技术迅猛发展的产物，也正是通过计算机，的企业才能更加方便的进行数据管理，科学化，正规化。让世界与中国相连。此次系统的开发也是为了解决现有房屋租赁市场存在的不规范的问题，让房屋管理员能够更加方便的管理日常房屋租赁事项。

## 1.2 研究现状

随着国内对于房屋租赁市场的更加重视，一些相关的政策出台，国的房屋租赁市场变得更加繁荣。虽然在疫情肆虐的影响下，2021年春节后，房屋租赁市场受到严重的影响，从年前销售万余套到如今的千余套，最低时更是不足300余套，但是随着天气回暖，疫情逐渐得到控制，房屋租赁市场的销售额逐渐提高，所以在现有的经济因素影响下，我国未来的房屋需求将会变得越来越大，市场规模也将变得越来越大。据估计，到2020年，国的房地产市场价格在国内预计达到5.71万亿美元，到2025年，国家的住房租金将达到9.82亿美元，这是几个世纪依赖不曾达到过的高度，租金居民将达到2.48亿美元。但是同时随着房地产产业的兴起，房地产市场也存在着很多问题 。

（1）地下租赁市场活动不受管控，由于疫情影响,社会经济收到了严重的打击，一些市场关停，外来务工人员找不到工作，而且房屋问题也影响着他们的日常生活，导致了一些不法分子趁虚而入，通过低价的出租房来诱骗外来务工人员，造成了大量的财产损失，不法中介也从中获利赚取高额中介费，对当前的房屋租赁市场产生了大量的不利影响。

　（2）国有土地大量流失。根据国家法规，为了个人利益，房屋所有者将通过规定必须向国家支付的限制，获得国家财产中已完工的土地的居住权。然而，实际上，由于经济上的利益，管理者避开了分支机构的管理，大量的住房流入地下黑市进行私人租赁。一些消息人士估计，大多数大城市的住房租金不足30%，造成了国家大部分财产收入损失。

（3）社会犯罪问题屡禁不止。由于管理失控，许多城市中的出租房被很多不法分子利用成为纳污藏垢之所，由近几年的资料显示，很多的刑事犯罪案件都与出租房有关，虽然有关组织以及政府把出租房房问题作为社会治安的重点问题，投入了很多的物理以及财力，许多的监管部门也加大了监管力度。但出租房犯罪活动任然猖獗，出租房犯罪问题依然不容小视。

所以在这种现状下，推出一个线上的租房订房网站就变得尤为重要，一方面他解决了出租房屋的繁琐性问题，易于房屋管理者在线上管理，也不用担心催缴房屋的问题，房屋管理者可以将自己空闲的房间放到网上进行出售，减轻了资金压力，也可以不用为高额的中介费而发愁.另一方面也方便了租客，网站上会显示房屋的具体信息，以及房屋图片,这样的话对于时间繁忙的租客就可以在短时间内找到自己心仪的房屋，大大减少了看房,买房所需的时间成本,租客可以有更多的时间去完成自己的一些工作，而不用为房屋问题而感到苦恼。并且在线上看房租房，可以减轻社会的犯罪率房屋管理者可以对租客的身份进行一个核实，如果租客的身份可能会导致犯罪，房屋管理者可以选择不将房屋租赁出去。提升社会租房的总体效率。

## 1.3 关键技术介绍

### 1.3.1 Java

Java语言是一种以面向对象为编程思想的语言，由Sun公司发布。Java语言的风格与C语言有着很多的类似，C语言是以面向过程为主要思想，而Java语言以面向对象为设计思想，独立开发的JVM，使其能够在Windows和Linux系统部署，从而让程序有了可移植性，简洁性，变得更加安全。而且Java语言与C语言相比，使用功能强大，学习起来更加轻松，对于初学的程序员来说，学习使用也没那么复杂。Java语言还具有很多特性，比如封装，继承，多态，这就在以面向对象为编程思想的同时，减轻了代码的复用率。让程序更加简介。

在Java语言发展了这么多年后，已经拥有了丰富的内库和成熟的开发解决方案，对于今后出现的复杂问题，Java语言也能轻松应对，所以在本系统选择以Java语言进行开发。

Java语言自诞生起就与其他语言有着些许不同，一种是面向编程的思想，另一种是面向对象的思想，而Java语言是面向对象的一种编程语言，Java语言在C++语言面向对象的基础上，增添了许多特性比如继承，多态等，放弃了指针和多继承，优化了内存空间的释放，闲置的内存空间的到释放，解决了程序员还需要管理内存空间的烦恼。Java语言还具有如下几个突出的特点：

（1）面向对象

Java语言的核心编程思想就是以面向对象为主，这一点让其与C语言和C++语言有着截然不同的区别，此外还具有继承，多态等特点。Java语言可以调用接口来解决

多继承问题。这次房屋租赁系统的实现使用的就是Java语言面向对象的特性，实现起来会更加简便方便。

（2）平台无关性

除此之外，Java语言还具有平台无关性，即Java程序的运行成功跟所使用的平台并没有太大关系，在Windows系统下写的程序在Linux下也是能够运行的。正是由此特点，所以Java程序具有其他语言没有的特点-可移植性。这样一来，Java程序就能通过虚拟机在不同的平台上在不需要编译的条件下就能运行。在这样的特性下，也是为Java语言在世界上流行奠定了基础，经历了这么多年的发展，Java语言的热度依然稳居世界榜首，成为众多程序员的不二选择。

（3）简单性

同样的之所以Java语言之所以能够在世界广受欢迎，同样离不开Java语言简单易学的特性，对于程序员而言，内存的管理是一件很头疼的问题，而Java语言提供的垃圾回收机制则恰好的解决好了这一问题。这样一来内存问题也得到了解决，程序员就可以将自己的全部精力投入到项目开发中去，大大缩短了开发时间与成本。

（4）多线程

Java语言还具有多线程这一特点，能够支持多个线程同时的进行，通过Thread类来创建，当然它的子类也可以创建。

### 1.3.2 Servlet

这次使用的Servlet技术，它是Web服务器和相关应用服务器的一种对应的程序，它的主要作用是作为中间层，收取来自HTTP客户端和Web服务器的请求。Servlet能够提供给的帮助有，可以通过Servlet收集来自前端传输过来的数据，从而显示来自相关库的资源和记录，还可以动态创建网页。

这次选择的Servlet技术在主流上依然是一个不错的选择，实现的效果上与CGI并没有太多区别。对于Servlet而言，它的优势还是很明显的，在性能方面，CGI的性能远没有Servlet好，能够更好的运用在web应用上。所以每个客户端的请求就能够同时处理，可以大大节省时间。Servlet跟Java语言一样，也具有平台独立性这一特点。为了确保计算机上的相关资源，在服务器上，相关的安全应用程序制定了一系列的限制条件。因此，Servlet 是可信的。Java 类库的全部功能对 Servlet 来说都是可用的。这样一来Servlet具有了Java语言的许多特性，为其在今后的发展奠定了基础。

Servlet（Server Applet）是JavaServlet的简称，称为小服务程序或服务连接器，Servlet这样具有交互式地接受前端传输的数据以及动态生成相关的web内容。比较片面的理解Servlet也可以认为他就是Java语言中的中的接口，而比较被大众所接受的理解是实现了这个接口的类。Servlet运行于支持Java的应用服务器中。如果探究其原理的话，它应该是可以对任意的一个类进行一个响应请求，但在使用的大多数情况下只用其来扩展服务器。最早支持Servlet标准的是JavaSoft JavaWebserver，此后，一些其它的基于Java的Web服务器开始支持标准的Servlet。

### 1.3.3 Jsp

一个系统能够被大众所接受，取决于其美观的动态刷新页面。在科技社会快速更迭的当下，数据的呈现方式变得丰富多彩，传统的网页技术无法满足管理员对数据呈现的要求，所以为了解决这一问题，可以通过使用JSP技术使其前端传输的页面来实现想要呈现的结果。

JSP来源于（Java Server Pages）是Sun Microsystems 公司，它的用途就是用来开发动态网页，是一种近几年十分受欢迎的动态网页实现技术，在web应用程序中的开发中用途广泛。在实质上，JSP与Servlet程序并没有什么区别，都是由Java语言编写的。在编写前端页面的过程中，为了呈现一些布局精巧的页面，就可以通过JSP技术来达到需要的效果，JSP在当前的web应用程序开发中非常受欢迎，在这次的房屋租赁管理系统中就多次用到了JSP页面，用来实现想要达到的效果。

### 1.3.3 JavaScript

JS的实质其实就是为了实现页面的动态效果，在使用过程中通常与CSS一起使用，来起到美化页面效果的作用，JS也是用Java语言编写，在语法使用上面并无太大差别。在用途方面，常用于Web应用程序开发。通过这种技术，可以在前端给用户呈现出一种布局精巧的体验感，优化细节处理，让用户更加乐意接受。如果想在页面中引入JavaScript可以通过标签的形式，也可以通过href在外部引入JavaScript文件也可以在HTML标签中的行内样式。

Javascript主要的用途以及相关的介绍

（1）通过JavaScript，可以实现前后端的交互

（2）JavaScript的实质是一种脚本语言，Java和Javascript是类似的。

（3）JavaScript一般用来编写客户端脚本。

（4）JavaScript同样还具有一个特点，那就是可以边执行边解释。

### 1.3.4 Jquery

Jquery是学习JavaScript需要熟练掌握的一个重要框架，他的出现是prototype之后又一个优秀框架，在2006年1月发布，它的设计使用宗旨就是想通过更少的代码来实现程序想要满足的业务。Jquery对JavaScript的代码块起到一个封装的作用，让广大程序员在设计JavaScript时可以更加简便，所以想要将前端技术学好,这是一项必须掌握好的技术。

JQuery 对象的由来就是通过Jquery通过包装DOM对象后所引出的对象。举个例子说明如下，可以 使用，$(“#exam”).Html，其中的意思是说：通过获取ID为Exam的元素内的Html代码。而HTML（）是JQuery里的代码，这段代码的作用可以认为用DOM实现**Document.getElementById(" test ").InnerHTML。** Jquery在今后的学习中占有非常重大的一部分，对于Jquery对象而言DOM对象里的方法是DOM对象独有的，即使Jquery对象来源于DOM，但是还是无法使用里面的方法，同理，如果DOM对象去引用Jquery里的方法，则结果会产生严重的错误，乱使用会报错。

## 1.4 开发工具和数据库的选择

### 1.4.1 IDEA

在这次开发工具的选择下，选择的是即可以写前端又可以写后台的IDEA，由JetBrains公司开发生产，主要用于Java语言方面的集成开发，是Java编程语言开发的集成环境。IDEA之所以用途如此广泛得益于在代码自动提示、重构、JavaEE支持、Java项目快速开发，在使用用途上远远超过市面上的Exlipse等开发工具，在其他方面，比如快速集成框架也是十分非常迅速高效的。IDEA还有其他的优点，IDEA在编写程序代码时也是十分严谨的，功能也是十分强大，比如快速生成代码，管理服务器页面，连接数据库等。IDEA总共有两个版本，免费版只支持少数语言，而旗舰版本中支持HTML，CSS,PHP等语言。

### 1.4.2 Tomcat

Tomcat服务器是相对轻量级的，而且用途广泛，通常用于中小型系统开发，这次的系统是相对中小型的，对于房屋租赁系统的实际需求来说是足够的，Tomcat是开发和调试

Web 应用的非常不错的选择。这次的房屋租赁管理系统中，采用的是Tomcat 9.0版本的来实现项目的搭建。

### 1.4.3 Mysql

大数据时代，传统关系型数据库管理系统(RDBMS)在处理海量数据时经历着严峻的性能考验。数据库性能的提升与数据查询的执行效率密切相关。查询操作是一个数据库 系统运行时最主要的负载。实上,对数据的增删改操作也常常建立在对所需数据检索的基础之上。查询语句是否迅速，快捷，在一定程度上决定了系统是否高效。

在关系型数据库中，Mysql数据库无疑是最突出的，与Oracle数据库中相比，Mysql是开源的一款数据库，可以高效的处理数据，对于第一次上手Mysql的程序员，操作也是十分简单。在这次的Mysql数据库中，也是采用了SQLyog可视化工具来创建这次项目中可能用到的用户表和房源表等等。

对于房屋租赁系统而言，数据库是否优良决定了系统是否高效，数据库是系统中存储用户数据的关键部分采用一个查询速度快，存储数据优良的数据库往往能决定系统的安全性，是系统是否稳定的关键部分。在这次的数据库设计中，使用Mysql数据库设计了房屋表，用户表，供应商表等等。

## 1.5 MVC 三层开发模式

在当前的web应用开发中，普遍采用的是MVC三层开发模式，分别是Model（模型层），View（视图层），Controller（控制层），在本次的房屋租赁系统开发中采用MVC三层设计模型可以实现业务层和视图层之间的代码分离，对于相同部分的程序代码就能按照不同的需求来展示不同的试图界面。

### 1.5.1 View

View(视图层)是用户与系统进行交互的一个页面，比如网页界面或者客户端界面,采用MVC三层架构，用户不需关注控制层和业务层之间代码是如何实现的，视图层中并没有对数据进行处理用户只需在视图层输入相关的指令，在控制层和业务层中处理数据后，就能呈现出相应的页面。

### 1.5.2 Model

Model(模型层)中是MVC三层架构中处理任务最多的一层，模型层与传入的数据并没有什么关系，这样一来，它就可以为多个试图提供支持，对于模型所用的代码只能在几个试图中重用一次，对于程序开发者来说比较友好，相关代码的重复率也就降低了，从而提高了开发的效率。

### 1.5.3 Controller

Controller(控制层) 是处理接受前端传入的参数或者请求，在业务层中调用相关的方法，查找数据库，将数据通过特定的视图来返回给用户。控制层本身并不做数据的产生和参数的处理，只是对传入的请求进行处理，调用业务层的相关方法来处理，最后通过试图的方式来返回给用户。

### 1.5.4 优点

（1）部署快，降低了维护成本。减少了程序开发的代码量以及使得维护用户截面处的成本降低，分离视图层和业务层，可以使开发人员对web应用的维护和维修修改成本降低，并提高效率的准确性。

（2）可扩展性增强。系统的可扩展性增强在未来产生新的需求时，只会修改对应的层的相关代码，整个三层机构并不会有太大的影响，节省了开发的工作时间。

（3）代码的复用性增强。MVC的设计模式能许可采用各式各样的试图页面来访问同一个web服务器上的程序代码，因为多个不同的试图能够同时想用一个业务逻辑的模型，而且返回的数据在模型层中没有格式化，所以具有相同特性的模型能够被不同的试图界面所使用，

（4）安全性能增加。三层架构的稳定性也使整个项目的安全性能上升。

### 1.5.5 缺点

降低了系统的性能实现起来也比较复杂，对于一些小型项目，本来是可以直接去访问数据库获取数据但是在三层架构的模式下，必须通过访问中间层才能完成，综合来看MVC三层设计架构对于大中型项目比较友好。

## 1.6 本文的组织结构

本文的内容结构编排如下：

（1）这次的房屋租赁系统论文，总共分为六章，分别是课题的研究背景，系统分析，总体设计，主要功能模块设计，系统测试与总结。

（2）在第1章，是介绍了当前房屋租赁系统这个课题当前的研究背景，以及当前的研究现状。为了更好的完成本系统，需要对当前课题进行深入的研究，来确定的研究课题是否有意义，在网上查阅了相关资料后，得出了当前房屋租赁市场的大致现状，这样就给的系统奠定了理论基础，从而继续接下来的研究。。

（3）第2章系统分析主要是阐述了房屋租赁系统的设计是否在当前条件下是否可行，主要分为技术可行性分析，资源可行性分析，风险可行性分析，功能需求分析和本次系统的用户角色，第二章在本篇论文占有很重的比列，因为只有确定了设计的系统是否可行，才能知道设计的系统是否有意义。

（4）第3章是总体设计，在确定好这次的设计总体思路后，总体设计中要确定设计的目标是什么，需要遵循怎样的设计原则，还有数据库该如何设计从而加快系统查询的速度。同时用户界面也要设计的精巧，让用户使用起来具有很好的体验感，同时设计系统的总体架构图，以及各个功能的模块架构图，只有设计出了骨架，才能对项目进行更加具体的划分。

（5） 第4章则是系统的主要模块设计实现，每个功能模块需要进行深入的研究后，才能将其付诸实践中去，主要分为会员注册模块,登录模块，房屋列表模块，个人信息及修改模块，后台管理模块。

（6）第5章的内容主要是对设计出来的房屋租赁系统进行系统测试，来检验是否达到了想要完成的目标，对第四章设计的主要模块进行测试。

（7）第6章是在这次房屋租赁系统的设计中对遇到的问题进行一个总结，总结自己在设计过程中遇到的问题，以及自己在设计中的思考的问题，检验自己在这次设计中考虑问题是否全面。

# 第2章 系统分析

## 2.1 可行性分析

一个项目是否能够通过检验，需要进行可行性分析。对一个项目进行细致的可行性分析可以加强将理论设计转变为实际产品的几率，对于一个没有进行过可行性分析的项目而言，产品失败的几率将大大升高，通过可行性分析，还可以在未来对于该系统是否可以继续发展进行具体的预测。

### 2.1.1 技术可行性分析

为了实现本系统，在设计细节处还得选择现在比较流行的技术，一个好的技术可以保证项目的成功，也能够满足未来的需求更新。在本次项目中选择的是Java语言编写，就本系统采用的技术而言，本系统主要是基于 Servlet作为控制台来对前端传输过来的信息进行一个整合。

### 2.1.2 资源可行性分析

从资源配置的角度来衡量该项目，可以清晰的预测该项目在实现企业经济发展目标、有效配置人力资源等方面将起到一个不错的效果。首先，Java语言已经在世界上发展了数十年，已经具有了成熟的开发能力，在一段时间内并不会被社会所淘汰；第二，当前系统采用的都是以Servlet为控制台，视图层，业务层三层架构来对前端传来的方法请求来进行处理。采用Tomcat部署web应用难度不会很大，简单方便，满足本次轻量级的应用开发成本也不会很高；最后，站在企业级开发的角度上来看，采用本系统来代替传统的房屋租赁方式，可以极大的减轻房屋管理员的工作量，能够长期保存数据，防止数据丢失。

### 2.1.3 风险可行性分析

一个项目的开发总有一些突如其来的风险产生，对一个项目提前进行风险可行性分析，采取相关的措施应对，可以极大减少不利因素的产生，来保证本系统可以长期的运行下去。在这次的房屋租赁系统中主要可能会产生这几点风险，首先：时间因素，本系统的开发要求能够在规定的时间内完成；第二：技术因素，选择的技术点是否能够支撑项目的开发，能够满足系统所需。第三：界面因素，前端展示的页面是否简介，清晰，能够被大众所接受。

### 2.1.4 功能需求分析

在对当前的研究课题进行实际分析后，们明确了系统需要实现的需求，接下来就是对房屋租赁系统的功能进行实际分析。

开发房屋租赁系统，首先需明确房东管理员和租客拥有的相关权限，租客可以在房屋租赁系统上查看房源信息，交付租金，联系房东，修改密码，房屋质量报修，申请看房等权限。房东管理员则拥有发布房源信息，催租，以及联系租客，处理房屋故障请求，修改密码等权限

租客在网站上首先需要查看房源信息，确认自己喜欢的房源信息后，点击提交订单来实现租房，当出现房屋质量问题可以提前告知房东，例如堆放物质量有问题的可以在网站上告知房东，如果需要提前退租可以在网站上点击提前退房，但也需要承担一部分的违约金。

房东管理员可以在网站上查看自己已经发布的房源信息，以及查看是否已经有租客确认订购，看房申请，收到需要处理的房屋故障报修也可以及时的处理，对于退租的消息也可以及时知晓。

## 2.2 用例图

本系统主要的人员有房东管理员以及租客，管理员的权限要比租客的权限大，他们之间所含有的权限用如下图2.1来表示

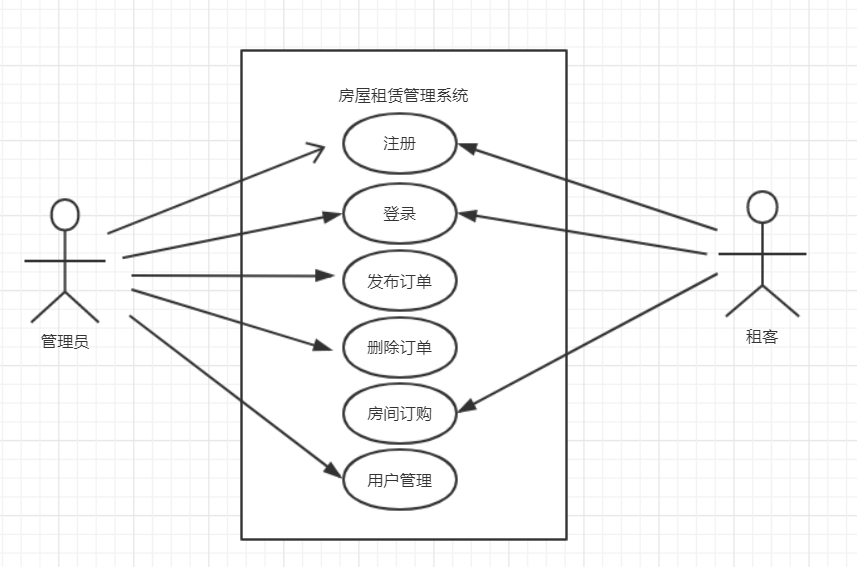


图2.1用例图

# 第 3章 总体设计

## 3.1 总体设计目标和设计原则

在明确了系统分析后，应该对系统设定一个总体的设计目标，对相关的数据库以及用户使用的界面等进行一个总体的规划，来确保项目的正确实施。

### 3.1.1 设计的总体思路

（1）首先准备好这次网站所需的静态资源，如前端页面，所用图形，maven项目所需要的环境以及依赖和一些Java环境配置，Idea配置，只有准备好了相关的工具后才能进行接下来的设计。

（2）完成注册功能模块能够进入网站首页，设计一个简洁大方的前端界面。

（3）完成注册功能后，实现登录功能，能够让刚刚注册好的用户名和密码能够登录到网站页面，在登陆时还应该区分用户角色，需要判断当前登录用户的角色是什么，是管理员还是租客。不同的角色所含有的功能也应该是不同的，权限也是不同的。

（4）完成登录功能后，还需要实现密码修改功能，当用户想要修改密码时，需要先输入原密码，如果原密码不对，则通过ajax方法来提示原密码出错，如果输入正确则提示原密码正确，之后输入新密码和确认密码，新密码和确认密码都应一样，否则提示无法修改密码，这样一来页面就可以在无刷新的情况下，及时给用户提供提示信息，也可以减轻服务器的工作量，在客户端就能解决。

（5）准备好房源列表模块和订单模块表，在设计房屋租赁系统时，为了防止数据库被破坏，维护数据库的安全性，对于不同角色所含有的权限也有所不同，在数据库设计时设计了租客和房东管理员两种了用户角色，1代表管理员，2代表租客。

### 3.1.2 总体设计目标

（1）简洁性。程序只有被大众所接受设计出来才有意义，房屋租赁系统的布局应该能够给用户一种清晰明了的感觉，用户能够直接在界面上找到自己想要的功能，系统界面还应该具有严谨的逻辑性，易于操作，不能轻率。

（2）可扩展性。系统不应满足当前的需求，还应该保持对未来需求的可扩展性，在设计系统时，可以预留相关的接口，在未来出现新的需求时，可以通过接口来实现。

（3）可靠性。系统还应具有可靠性，对于外部因素或内部性因素造成系统不稳定或者死机的情况出现，系统能够触发的相关机制使系统尽快恢复，也能保障系统数据不被丢失。即使数据丢失了，也要有相关的措施找回。

（4）兼容性。系统开发成功后，不可能只在一个操作系统或者浏览器上运行，所以还应保证系统具有通用性，能够满足不同用户在不同系统上的运行需求，不定期的对系统兼容性进行优化。

（5）安全性。一个系统能够长久的运行下去，安全性也是必不可少的一个因素。对于用户权限应该严格进行管控，避免出现删库跑路的现象出现，也要定期对数据进行备份，防止出现数据丢失对系统造成的不利结果出现。

### 3.1.3 系统设计原则

（1）系统性原则。系统应始终从总体设计目标出发，服从总体设计要求，局部服从全局，保证系统设计的方案成为一个有机的整体。

（2）实用性原则。设计的系统应该以满足用户需求为出发点，只有一个系统具有实用性才能被大众接受。对于房东管理员，租客两种角色用户，他们具有的权限，功能也应是不同的，不同的业务功能需求也可以通过不同的方法来满足。

（3）可靠性原则。选择的技术应该要经过研究分析后得出，要尽量满足开发的需求。选择技术时，应该选择当下流行的技术，最好能够被市场所接受，在以后的开发过程中也应该具有高维护性，出现的需求能够及时满足。

（4）可维护性和可扩展性原则。在进行模块设计时，应对遵照模块独立性的思想，高内聚低耦合应对使设计程序时应当考虑的首要原则之一，对今后出现的需求也应能够高效的处理，出现问题时维护起来更加方便。

### 3.1.4 数据库设计原则

（1）数据安全性原则。对于任何一种管理系统来说，数据的安全性往往占据了重大的地位，数据如果泄露的话会造成不可估计的财产损失，所以在设计数据库时，要充分对用户的权限进行筛选，防止不法分子利用数据库泄露用户的相关隐私，造成财产损失。

（2）数据一致性原则。在设计数据库时，要使得房屋租赁的实际需求与数据库中的表具有一一对应的关系，要保证数据库中的数据与实际需求相对应。在前面几节对真实条件下用户的实际需求进行分析后，对每个用户设计一张数据库表，来保证数据的一致性。对每个事件产生后出现的新数据与原始数据间的对应关系应该保持一致。在操作数据库操作时，要严格遵守数据库的使用原则，防止出现数据不一致的现象。

（3）数据的完整性原则。数据在存储时应该是完整的，为了防止出现数据不匹配的情况。在系统的开发过程中，要对表中的数据进行严格处理，比如外键，规则等，从而保证项目能够成功实现。

（4）数据规范化原则。对于房屋租赁系统而言，数据是用户资料的表达方式，数据在本系统中占有者很高的地位。设计过程中要严格以规范化为设计原则，从而使数据更加精确，如果数据模型设计的不规范则可能会造成数据丢失或者数据异常的情况出现，这些情况造成的后果都是不可预估的，在实际开发中，要尽量避免这些情况出现。

（5）数据可扩展性原则。房屋租赁本身就是方式多变的，在设计过程中，要考虑在市场更新时可能会造成的数据迁移，升级的现象出现，所以数据也应该与时俱进，具有可扩展性。对于数据而言，还要满足今后的市场所需，应具有一定的冗余能力。

### 3.1.5 用户界面设计原则

（1）用户至上设计原则。用户对一个产品的喜爱往往是第一印象的好坏，同样的，一个系统研究出来是否有意义取决于用户对产品的接受程度，功能十分强大但设计界面潦草的系统不堪是不会被大众所青睐。所以在设计前端页面时，要充分考虑用户对本系统的看法，设计出一个交互能力强，布局精巧，视觉程度感拉满的前端页面。只有被大众所接受，接下来的开发才能继续下去。

（2）简易性设计原则。从近几年的研究来看，青年人对于简介的事务接受能力要强。所以在设计房屋租赁系统时要对用户界面的设计更加具有简洁性，层次性，设计一些关键的导向性图标，能够在第一时间内满足用户的需求，从而在事务的处理过程中，体现出本业务的核心逻辑。

### 3.1.6 系统架构图

在对前面几节的分析中，对于房屋租赁系统而言得出实现房屋租赁管理系统总共需要三个部分，分别是用户层，Servlet层以及后台数据处理层，如下图3.1。

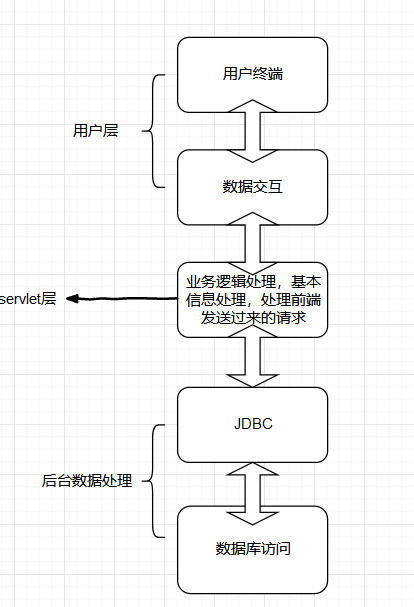


图3.1系统架构图

分层架构可以解决代码工作量的问题，设计房屋租赁系统时将总体架构分为三个层次，层次之间通过调用相应的接口进行通信，这样设计的好处是，当某个层需要更新用户传输进来的新需求时，只需更新当前层的相关代码，接口可以保持不变这样就能节省代码的工作量，而且在日后的维护中也会相当方便，对于一个需要处理大量信息的系统来说，这样的设计是非常有必要的。下面来分析每一个层次的具体内容。

（1）用户层

设计用户层的目的是为了让用户与系统之间可以更好的交互，系统提供简介强大的前端界面，用户可以通过前端界面来输入自己的命令，当然死板的用户界面无法引起用户的兴趣，通可以过AJAX,JSP，CSS等技术来让用户有一个更好的体验感。

用户输入的命令通过Servlet层处理后实时的呈现在前端界面中，用户不需要了解数据如何呈现在前端页面，只需关注数据呈现的是否准确。

（2）Servlet层

这一层介于用户层和后台处理层之间，一方面需要快速响应用户的请求，另一方面需要快速请求数据，由此观之，对响应时间的要求也非常高。另外，由于多个管理子过程在运行时，可能由于资源冲突出现阻塞延迟的现象，所以务必要对此进行优化，避免系统崩溃。

用户发送过来的数据需要在Servlet层进行处理，所以这一层在三层架构里面占据最重要的一部分。对于房屋管理员而言，这一层控制着房屋的资料，租金等敏感信息，是绝对不能出错的一部分，为了保住正确性，这就要求Servlet层算法的速度要够快，够高效，

这样才能满足用户的基本需求，防止出错；兼容性也要好，当出现不同的用户请求时可以高效的处理。

（3）后台处理层

后台处理层的作用是对Servlet层发送过来的请求来调用数据库里面的数据进行一个处理，数据库在后台处理层中占据着很大的地位，当Servlet请求对数据库的系统进行调用时，只能通过相应的接口来执行sql命令，最后通过相关的数据访问组件访问数据库。这样一来，数据就不需要直接被系统应用调用，具有很强的独立性，减少了系统崩溃的几率，同时增强了系统的可移植性。

### 3.1.7 功能模块结构图

主要的功能模块如下，如图3.2

（1）会员注册板块：使用电话号码注册，验证码检验，出错时提示；

（2）登录板块：注册的电话号码登录，出错时提示；

（3）房屋列表板块：根据类别、价格等进行排序；

（4）房屋详情页面：房屋文字图片信息；

（5）个人信息及修改：对用户资料进行查看和修改；

（6）后台管理：各级管理员负责的网站信息维护。

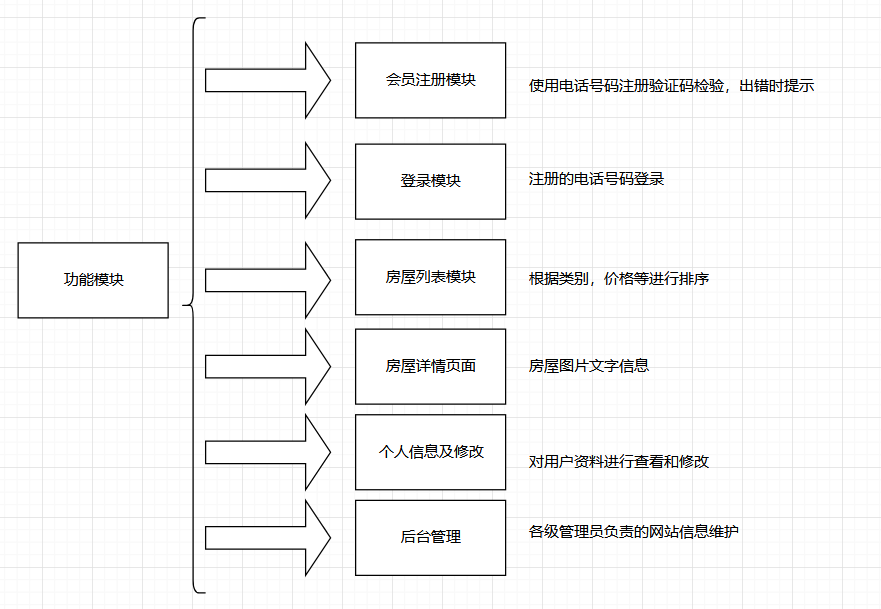


图3.2功能模块图

## 3.2 数据库设计

在这次的房屋租赁系统开发中选择Mysql数据库进行开发，先是新建了smbms的数据库，再在数据库中建立这次会用到的相关表，如用户表，房源表，供应商表等，采用SQLyog可视化工具进行开发，下面的这些表就是系统开发中可能会用到的数据表。

表3.1 用户表:smbms-user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 列名 | 数据类型 | 长度 | 描述 |
| 编号 | Id | Int | 8 | 主键、非空、自增 |
| 用户编码 | userCode | Varchar | 15 | 用于登录 |
| 姓名 | userName | Varchar | 15 | 非空， |
| 密码 | userPassword | Varchar | 15 | 用于用户登录的密码 |
| 性别 | gender | Int | 10 | 1：女 2：男 |
| 出生日期 | birthday | Varchar | 15 | 用户的出生日期 |
| 电话号码 | phone | Varchar | 13 | 纯数字字符 |
| 家庭住址 | adress | Varchar | 12 | 按照省市区的格式保存 |
| 角色 | userRole | Int | 20 | 1-系统管理员2-普通租客 |
| 创建时间 | creationDate | Datetime |  | 角色创建的时间 |
| 被谁更新 | modifyBy | Int | 10 | 通过谁更新了数据 |
| 更新时间 | modifyDate | Datetime |  | 更新的时间 |

表3.2用户角色表:smbms-role

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 |  | 数据类型 | 长度 | 描述 |
| 编号 | Id | Int | 8 | 主键、非空、自增 |
| 角色编码 | userCode | Varchar | 15 | 角色的编码表示 |
| 角色姓名 | roleName | Varchar | 15 | 非空 |
| 被谁创建 | CreatedBy | Int | 10 | 通过谁更新了数据 |
| 创建时间 | Createtime | Datetime |  | 更新的时间 |

表3.3 房屋供应商表：smbms-provider

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 列名 | 数据类型 | 长度 | 备注 |
| 编号 | Id | Int | 8 | 主键、非空、自增 |
| 供应商编码 | proCode | Varchar | 15 | 供应商编码 |
| 供应商姓名 | proName | Varchar | 15 | 供应商姓名 |
| 供应商描述 | proSec | Varchar | 15 | 供应商描述 |
| 负责人 | proContact | Varchar | 10 | 负责人 |
| 电话号码 | proPhone | Varchar | 13 | 纯数字字符 |
| 供应商住址 | proAdress | Varchar | 12 | 按照省市区的格式保存 |
| 供应商传真 | proFax | Int | 20 | 供应商传真 |
| 被谁创建 | createdBy | Int | 8 | 被谁创建（1：管理员 2：供应商） |
| 创建时间 | creatTime | DateTime |  | 创建时间 |

表3.4订单表:smbms-bill

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名名称 | 列名 | 数据类型 | 长度 | 备注 |
| 编号 | Id | Int | 8 | 主键、非空、自增 |
| 订单编码 | userCode | Varchar | 15 | 角色的编码表示 |
| 房源名称 | roleName | Varchar | 15 | 房源名称 |
| 供应商名称 | Proname | Varchar | 10 | 通过谁更新了数据 |
| 是否租赁 | IsPayment | Int | 8 | 1:已出租 2：未出租 |
| 租金 | Count | Int | 8 | 房源租金 |
| 创建时间 | Createtime | DateTime |  | 创建时间 |

表3.5 供应商表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 列名 | 数据类型 | 长度 | 备注 |
| 编号 | Id | int | 8 | 主键、非空、自增 |
| 房源编码 | userCode | Varchar | 15 | 角色的编码表示 |
| 房源名称 | roleName | Varchar | 15 | 房源名称 |
| 供应商名称 | Proname | Varchar | 10 | 房源供应着 |
| 是否租赁 | IsPayment | int | 8 | 1:已出租 2：未出租 |
| 租金 | PayCount | Int | 8 | 房源租金 |
| 房源数量 | count | int | 8 | 房源数量 |
| 创建时间 | Createtime | DateTime |  | 创建时间 |

# 第4章 主要功能模块的设计与实现

## 4.1会员注册功能

首先访问一个网站最基本的就是先要有一个账号，对于一个普通租客来说需要的就是先申请一个账号，前端传输出来的数据出来的数据通过ajax局部校验，如果数据规范，那么就将其数据写入数据库，最后返回到登录页面实现登录操作，如果要是用网站的其他功能，那么注册功能是一个网站功能中最基本也是最重要的功能。

会员注册的流程图如下图4.1

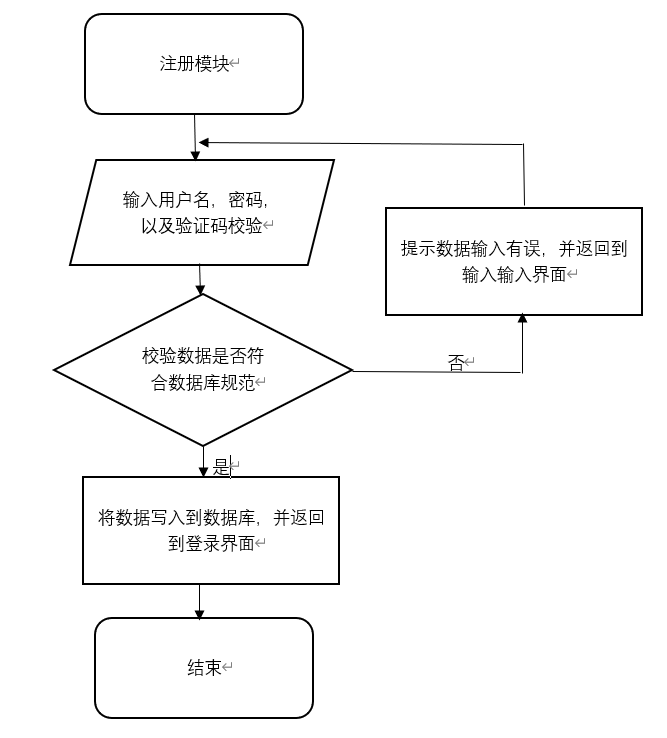


图4.1注册流程图

### 4.1.1 主要代码实现

public boolean add(User user) {

// TODO Auto-generated method stub

boolean flag = false;

Connection connection = null;

try {

connection = BaseDao.getConnection();

connection.setAutoCommit(false);//开启JDBC事务管理

int updateRows = userDao.add(connection,user);

connection.commit();

if(updateRows > 0){

flag = true;

System.out.println("add success!");

}else{

System.out.println("add failed!");

}

} catch (Exception e) {

try {

System.out.println("rollback==================");

connection.rollback();

} catch (SQLException e1)

e1.printStackTrace();

}

}finally{

//在service层进行connection连接的关闭

BaseDao.closeResource(connection, null, null);

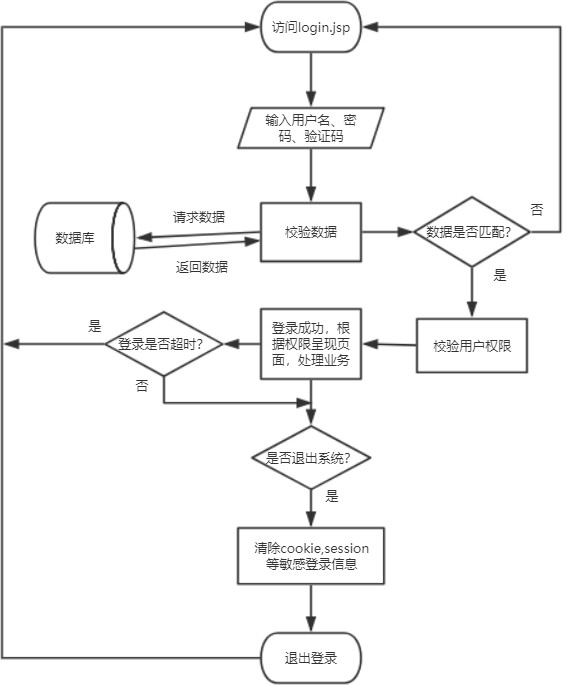
}

return flag;

}

## 4.2 登录模块

账号创建成功后就要实现登录操作了，一开始用户如果想要访问网站页面的话，就需要先通过创建的账号和密码进行登录，从而访问login.jsp，login.jsp进行数据校验，通过访问数据库，校验传输进去的数据是否与数据库中的一致进行数据校验，如果没有通过的话就提示用户输入的数据有误，请重新输入，之后校验用户角色是管理员还是普通用户。登录成功后根据角色权限显示相关页面。退出系统后也需要清除用户的session，防止不法分子访问网站内的重要页面以及资源，流程图如下图4.2。

 图4.2登录流程图

### **4.2.1** 主要代码实现

Login.jsp

这里主要是通过表单提交到后台的一部分代码

<body class="login\_bg">

<section class="loginBox">

<header class="loginHeader">

<h1>房屋租赁管理系统</h1>

</header>

<section class="loginCont">

<form class="loginForm" action="${pageContext.request.contextPath }/login.do" name="actionForm" id="actionForm" method="post" >

<div class="info">${error }</div>

<div class="inputbox">

<label>用户名：</label>

<input type="text" class="input-text" id="userCode" name="userCode" placeholder="请输入用户名" required/>

</div>

<div class="inputbox">

<label>密码：</label>

<input type="password" id="userPassword" name="userPassword" placeholder="请输入密码" required/>

</div>

<div class="subBtn">

<a href="${pageContext.request.contextPath }/registered.jsp">注册</a>

</div>

</form>

</section>

</section>

</body>

service层：从数据库获取到的user，判断是否为空

public User login(String userCode, String userPassword) {

// TODO Auto-generated method stub

Connection connection = null;

User user = null;

try {

connection = BaseDao.getConnection();

user = userDao.getLoginUser(connection, userCode);

} catch (Exception e) {

e.printStackTrace();

}finally{

BaseDao.closeResource(connection, null, null);

}

//匹配密码

if(null != user){

if(!user.getUserPassword().equals(userPassword))

user = null;

}

return user;

}

Servlet层：调用reques方法获取service层传输进来的参数，如果参数不为空则登录成功，并实现页面的跳转。

System.out.println("login ============ " );

//获取用户名和密码

String userCode = request.getParameter("userCode");

String userPassword = request.getParameter("userPassword");

//调用service方法，进行用户匹配

}

## 4.3 房屋列表模块

房屋列表模块主要是展现已经创建好的房源信息，并且提供查询功能，以及对房源信息进行修改以及删除等功能，房源列表流程图如下。

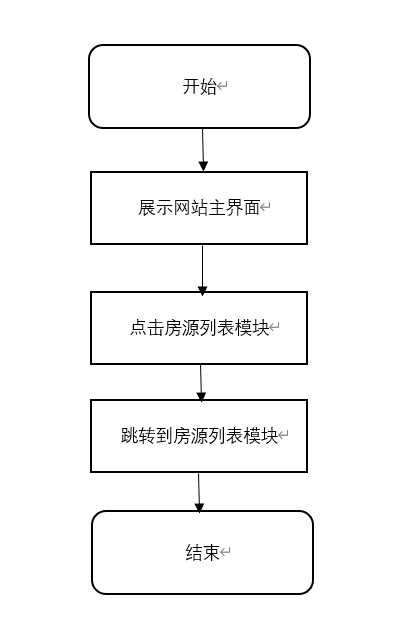


图4.3 房源列表流程图

### 4.3.1 主要代码实现

在service层展示房屋列表，通过获取用户session.user中来判断用户当前的角色

<input value="查 询" type="submit" id="searchbutton">

<a href="${pageContext.request.contextPath }/jsp/billadd.jsp">添加订单</a>

</form>

</div>

<!--账单表格 样式和供应商公用-->

<table class="providerTable" cellpadding="0" cellspacing="0">

<tr class="firstTr">

//创建订单编码，房源名称，供应商，租金金额，是否租赁，创建时间，操作等 </tr>

<c:forEach var="bill" items="${billList }" varStatus="status">

<tr>

<td>

<span>${bill.billCode }</span>

</td>

<td>

<span>${bill.productName }</span>

</td>

<td>

<span>${bill.providerName}</span>

</td>

<td>

<span>${bill.totalPrice}</span>

</td>

<td>

<span>

<c:if test="${bill.isPayment == 1}">未出租</c:if>

<c:if test="${bill.isPayment == 2}">已出租</c:if>

</span>

</td>

<td> <span>

<fmt:formatDate value="${bill.creationDate}" pattern="yyyy-MM-dd"/>

</span>

</td>

<td>

<span><a class="viewBill" href="Javascript:;" billid=${bill.id } billcc=${bill.billCode }><img src="${pageContext.request.contextPath }/images/read.png" alt="查看" title="查看"/></a></span>

<span><a class="modifyBill" href="Javascript:;" billid=${bill.id } billcc=${bill.billCode }><img src="${pageContext.request.contextPath }/images/xiugai.png" alt="修改" title="修改"/></a></span>

<span><a class="deleteBill" href="Javascript:;" billid=${bill.id } billcc=${bill.billCode }><img src="${pageContext.request.contextPath }/images/schu.png" alt="删除" title="删除"/></a></span>

</td>

</tr>

</c:forEach>

</table>

</div>

</section>

<!—删除后的提示页面 -->

<div class="zhezhao"></div>

<div class="remove" id="removeBi">

<div class="removerChid">

<h2>提示</h2>

<div class="removeMain">

<p>你确定要删除该订单吗？</p>

<a href="#" id="yes">确定</a>

<a href="#" id="no">取消</a>

</div>

</div>

</div>

## 4.4 信息修改页面

模块的功能主要是检索自己已经填写的信息，以及修改自己的个人密码，输入密码首先通过ajax来判断写入的密码是否符合要求规范，如果不符合规范则提示密码输入不合法，如果正确就将其写入数据库中，将数据库中的密码进行更新，如下图4.3。

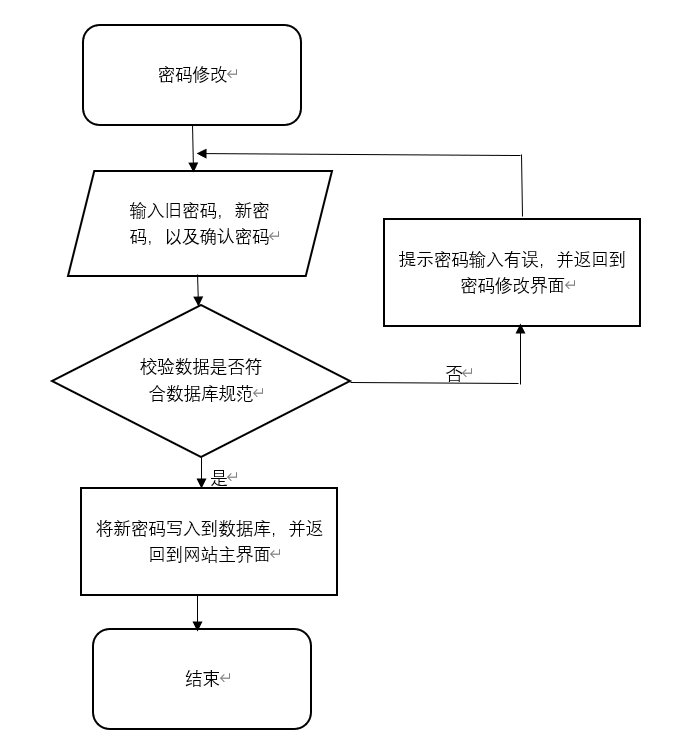


图4.4 信息修改流程图

### 4.4.1主要代码实现

密码修改的jsp页面

前端输入密码后，在pwdmodify.js中进行规范性判断。

var oldpassword = null;

var newpassword = null;

var rnewpassword = null;

var saveBtn = null;

$(function(){

oldpassword = $("#oldpassword");

newpassword = $("#newpassword");

rnewpassword = $("#rnewpassword");

saveBtn = $("#save");

oldpassword.next().html("\*");

newpassword.next().html("\*");

rnewpassword.next().html("\*");

oldpassword.on("blur",function(){

$.ajax({

type:"GET",

url:path+"/jsp/user.do",

data:{method:"pwdmodify",oldpassword:oldpassword.val()},

dataType:"json",

successive:function(data){

//如果用户输入的 旧密码正确

validateTip(oldpassword.next(),{"color":"green"},imgYes,true);

}else if(data.result == "false"){//旧密码输入不正确

validateTip(oldpassword.next(),{"color":"red"},imgNo + " 原密码输入不正确",false);

}else if(data.result == "sessionerror"){//当前用户session过期，请重新登录

validateTip(oldpassword.next(),{"color":"red"},imgNo + " 当前用户session过期，请重新登录",false);

}else if(data.result == "error"){//旧密码输入为空

validateTip(oldpassword.next(),{"color":"red"},imgNo + " 请输入旧密码",false);

}

},

error:function(data){

//请求出错

validateTip(oldpassword.next(),{"color":"red"},imgNo + " 请求错误",false);

}

});

}).on("focus",function(){

validateTip(oldpassword.next(),{"color":"#666666"},"\* 请输入原密码",false);

});

newpassword.on("focus",function(){

validateTip(newpassword.next(),{"color":"#666666"},"\* 密码长度必须是大于6小于20",false);

}).on("blur",function(){

if(newpassword.val() != null && newpassword.val().length > 6

&& newpassword.val().length < 20 ){

validateTip(newpassword.next(),{"color":"green"},imgYes,true);

}else{

validateTip(newpassword.next(),{"color":"red"},imgNo + " 密码输入不符合规范，请重新输入",false);

}

});

rnewpassword.on("focus",function(){

validateTip(rnewpassword.next(),{"color":"#666666"},"\* 请输入与上面一致的密码",false);

}).on("blur",function(){

if(rnewpassword.val() != null && rnewpassword.val().length > 6

&& rnewpassword.val().length < 20 && newpassword.val() == rnewpassword.val()){

validateTip(rnewpassword.next(),{"color":"green"},imgYes,true);

}else{

validateTip(rnewpassword.next(),{"color":"red"},imgNo + " 密码输入不一致，需要重新输入",false);

}

});

saveBtn.on("click",function(){

oldpassword.blur();

newpassword.blur();

rnewpassword.blur();

if(oldpassword.attr("validateStatus") == "true"

&& newpassword.attr("validateStatus") == "true"

&& rnewpassword.attr("validateStatus") == "true"){

if(confirm("确定要修改密码？")){

$("#userForm").submit();

}

}

});

});

# 第 5章 系统测试

## 5.1 会员注册功能

通过前端输入的密码以及账号来判断是否符合规范性要求，在js中使用ajax来进行异步处理节省返还给数据库的时间，如果符合规范性要求，就将其发还给数据库，测试结果如图5.1。



图5.1注册图

## 5.2 登录功能

注册功能完成后，就可以通过注册的账号和密码登录网站主界面，测试结果如图5.2



图5.2登录实现图



图5.3注册成功截图

## 5.3 房屋列表模块

该页面用来展示当前可以订购的房源有多少，不同的用户角色所具有的权限也是不同的，测试结果如图5.4。



图5.4 房屋列表图

## 5.4 订单详情模块

在订单管理模块中可以查询与自己有关的相关订单，测试结果如图5.5图图5.5 订单详情图

## 5.5 信息修改页面

个人相关信息及修改模块主要从事的是对个人的密码进行一个修改，修改成功后，会将修改成功的密码写入到数据库，测试结果如图5.6



图5.6密码修改截图

## 5.6 发布订单页面

订单发布页面是管理员发布订单的功能，可以为用户提供房源



图5.6 发布订单页面

## 5.7 添加供应商页面

添加房屋供应商



图5.7 房屋供应商

# 第6章 总结

随着现代社会科学发展的速度变得越来越快，如何满足一些年轻人，以及大学生毕业的房屋需求变得越来越重要，互联网的飞速发展使得可以在网络上进行房屋的租赁以及购买，研发出一个能够满足现在人租房需求的网站是当前的一个重要课题。

这次的毕业设计是一个不小的考验，即将毕业的要交出一份完美的论文试卷来回馈在湖工的这四年时光，导师这次的毕业论文题目是要求完成房屋租赁系统的设计与实现这个课题，觉的这是一份很好的论文题目，对于其中登录模块，注册模块，以及订单管理模块刚开始不知道该怎么写，到之后查阅资料才慢慢动手，对于经济科学文化快速发展的今天，大学生毕业后需要租房的压力变得越来越大，现在的房屋信息更新并不及时，一些房屋需要去现场进行看房，如果有一个能够在线解决看房的问题的网站的话，这将极大节省的时间。

在这次毕业设计中,实际上也是将4年学到的知识进行了综合的应用,使更进一步掌握和了解了如何通过自己深入的调研，编程，设计来完成-一个系统的开发。在这次的毕业设计中，对MVC三层架构和相关的编程知识以及房屋租赁的市场了解更深了，对今后的职场生涯也有了更加清晰的认识，但是由于自己能力有限，经验不足，是的本系统还有很多不足以及需要完善的地方，还需要老师和同学们的指导，改正。

房屋租赁管理系统在公租房信息化管理占有很重要的一部分。随着社会经济和科学技术的快速发展、居民住房的需求增加改善和外来务工人员对房屋的需求增加人，出租房屋数量剧增。